Algoritmos y Estructuras de Datos

Cursada 2024

Grafos

Caminos de costo mínimo

- Data Structures and Algorithm Analysis in Java; 2nd Ed. Mark Allen Weiss (Capítulo 9)
- Estructuras de datos y algoritmos; Mark Allen Weiss. (Capítulo 9)

Caminos mínimos entre todos los pares de vértices

- Estrategia: Algoritmo de Floyd
  - Lleva dos matrices D y P, ambas de |V| x |V|

Matriz de costos mínimos

Matriz de vértices intermedios

El costo total del algoritmo es $O(|V|^3)$

Algoritmo de Floyd

Camino de costo mínimo entre cada par de vértices

```
para i=1 hasta cant_Vértices(G)
        para j=1 hasta cant_Vértices(G)
            D[i,j] = A[i,j]

para k=1 hasta cant_Vértices(G)
      para i=1 hasta cant_Vértices(G)
        para j=1 hasta cant_Vértices(G)
            si (D[i,j] > D[i,k] + D[k,j]) {
                 D[i,j] = D[i,k] + D[k,j];
                P[i,j] = k;
             }
```

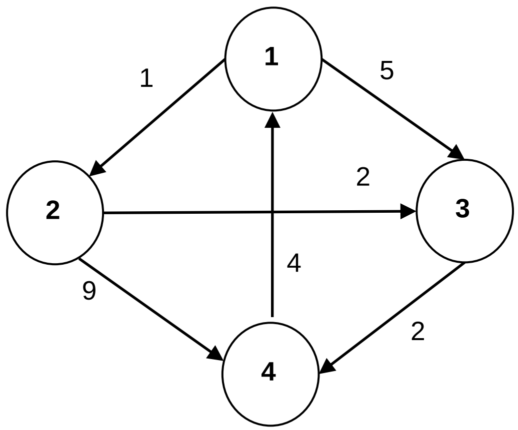
Toma cada vértice como intermedio, para calcular los caminos

Distancia entre los vértices *i* y *j*, pasando por *k*

Algoritmo de Floyd

Camino de costo mínimo entre cada par de vértices

➢ Ejemplo:



$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	∞	∞	0

K=1

$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	~~∞~~ **5**	~~∞~~ **9**	0

Poner otros ejemplos

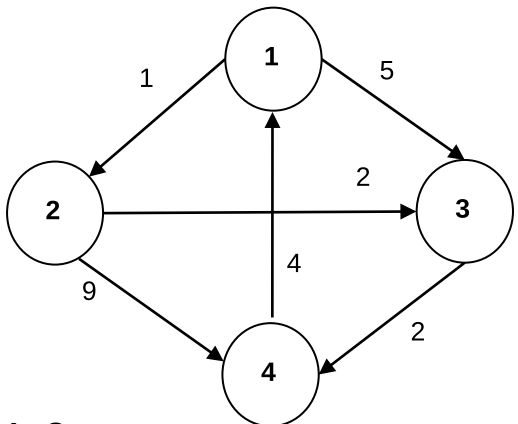
$$D(4,2) > D(4,1) + D(1,2) \rightarrow D(4,2) = D(4,1) + D(1,2)$$
$$\infty > 4 + 1 \rightarrow D(4,2) = 5$$

$$D(4,3) > D(4,1) + D(1,3) \rightarrow D(4,3) = D(4,1) + D(1,3)$$
$$\infty > 4 + 5 \rightarrow D(4,3) = 9$$

Algoritmo de Floyd

Camino de costo mínimo entre cada par de vértices

- Ejemplo:



$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	∞	∞	0

K=1

$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	<u>5</u>	<u>9</u>	0

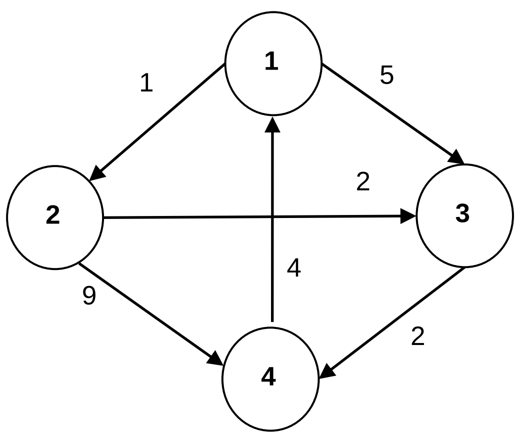
K=2

$D_{i,j}$	1	2	3	4
1	0	1	~~5~~ **<u>3</u>**	~~∞~~ **<u>10</u>**
2	∞	0	2	9
3	∞	∞	0	2
4	4	<u>5</u>	~~9~~ **<u>7</u>**	0

Algoritmo de Floyd

Camino de costo mínimo entre cada par de vértices

- Ejemplo:



$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	∞	∞	0

K=1

$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	5	9	0

K=2

$D_{i,j}$	1	2	3	4
1	0	1	3	10
2	∞	0	2	9
3	∞	∞	0	2
4	4	5	7	0

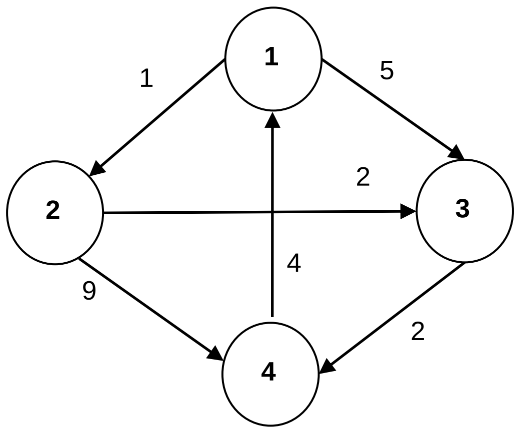
K=3

$D_{i,j}$	1	2	3	4
1	0	1	3	~~10~~ **5**
2	∞	0	2	~~9~~ **4**
3	∞	∞	0	2
4	4	5	7	0

Algoritmo de Floyd

Camino de costo mínimo entre cada par de vértices

- Ejemplo:



$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	∞	∞	0

K=1

$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	5	9	0

K=2

$D_{i,j}$	1	2	3	4
1	0	1	3	10
2	∞	0	2	9
3	∞	∞	0	2
4	4	5	7	0

K=3

$D_{i,j}$	1	2	3	4
1	0	1	3	5
2	∞	0	2	4
3	∞	∞	0	2
4	4	5	7	0

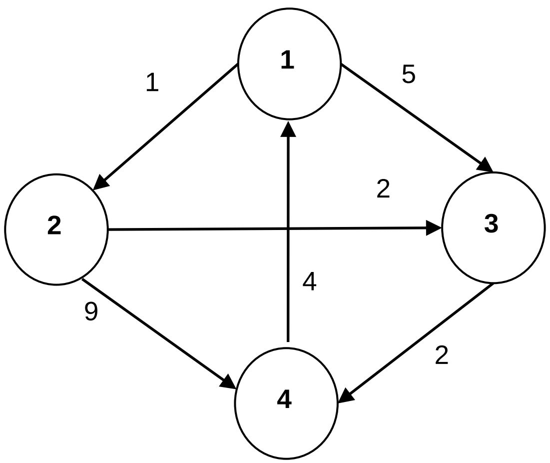
K=4

$D_{i,j}$	1	2	3	4
1	0	1	3	5
2	~~∞~~ **8**	0	2	4
3	~~∞~~ **6**	~~∞~~ **7**	0	2
4	4	5	7	0

Algoritmo de Floyd

Camino de costo mínimo entre cada par de vértices

- Ejemplo:



$D_{i,j}$	1	2	3	4
1	0	1	5	∞
2	∞	0	2	9
3	∞	∞	0	2
4	4	∞	∞	0

Matriz inicial de costos
entre cada par de vértices

$D_{i,j}$	1	2	3	4
1	0	1	3	5
2	8	0	2	4
3	6	7	0	2
4	4	5	7	0

Matriz luego de aplicar
Floyd con los costos
entre cada par de vértices